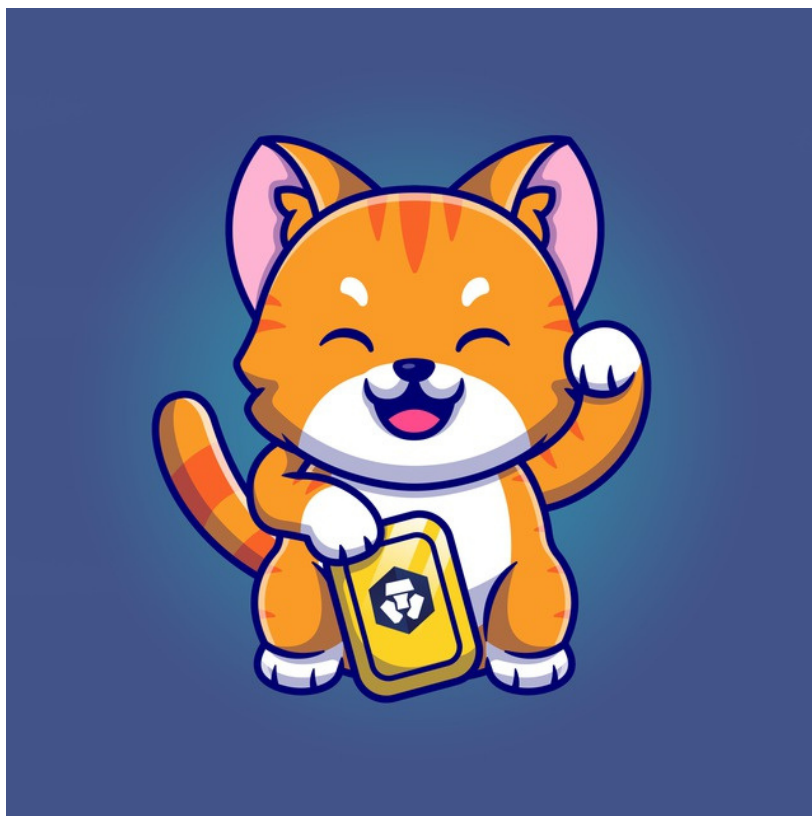




# PREMIUM AUDIT

|                    |  |
|--------------------|--|
| NAME OF PROJECT    | Crocate                                    |
| TOKEN CRO ADDRESS  | 0xa88fb0aabd0f0aa61cff3853adbc0b6e49a8e2f1 |
| WEBSITE / TELEGRAM | crocate.com                                |



**DISCLAIMER: PLEASE READ FULL AUDIT**



# IMPORTANT DISCLAIMER

---

I.

## **NOT RESPONSIBLE**

Analytix Audit holds no responsibility for any actions from the project in this audit.

II.

## **NOT GUARANTEE**

Analytix Audit in no way guarantees that a project will not remove liquidity, sell off team tokens, or exit scam.

III.

## **INFORMATION**

Analytix Audit researches and provides public information about the project in an easy-to-understand format for the common person.

IV.

## **AUDIT AGREEMENT**

This audit agreement does not guarantee ANY illicit actions by the project team and does not serve as an advocacy for the project.

V.

## **NO ADVICE**

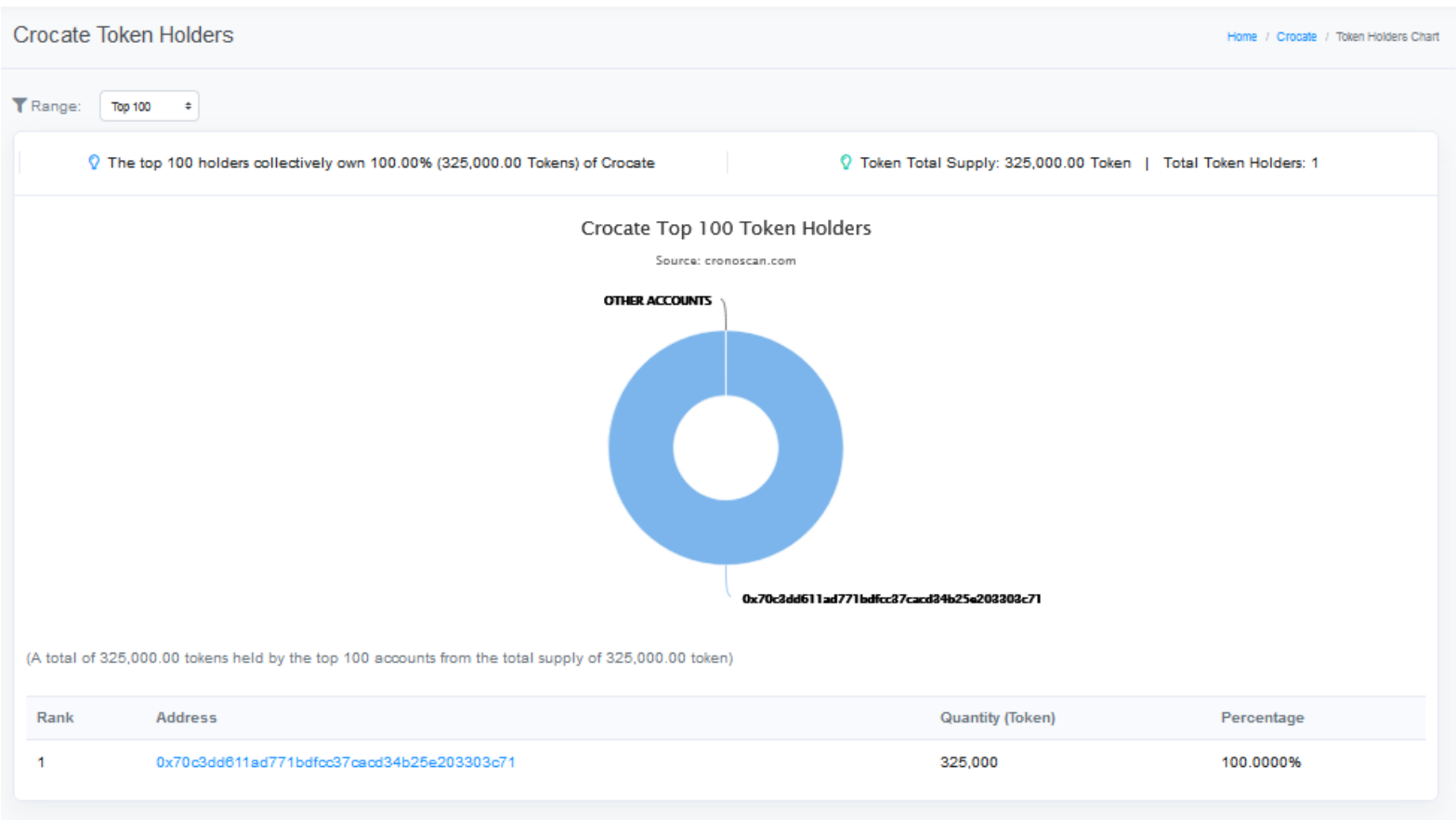
Analytix Audit in no way takes responsibility for any losses, not does Analytix Audit encourage any speculative investments.

VI.

## **DISCLAIMER**

The information provided in this audit is for information purposes only and should not be considered investment advice. Analytix Audit does not endorse, recommend, support, or suggest any projects that have been audited.

# TOKENOMICS - OVERVIEW



- More than 15% of the tokens are unlocked

(This comment refers to when the audit was carried out maybe team locked tokens later)

# CONTRACT CODE - OVERVIEW



## QUICK REVIEW

- Owner can set fees up to 35%
- Owner can set tx Limits  
(minium 0.1% supply)

# Owner Privileges:

```
520 * Requirements:
521 *
522 * - the calling contract must have an ETH balance of at least `value`.
523 * - the called Solidity function must be `payable`.
524 *
525 * _Available since v3.1._
526 */
527 function functionCallWithValue(
528     address target,
529     bytes memory data,
530     uint256 value
531 ) internal returns (bytes memory) {
532     return functionCallWithValue(target, data, value, "Address: low-level call with value failed");
533 }
534
535 /**
536 * @dev Same as {xref-Address-functionCallWithValue-address-bytes-uint256}[`functionCallWithValue`], but
537 * with `errorMessage` as a fallback revert reason when `target` reverts.
538 *
539 * _Available since v3.1._
540 */
541 function functionCallWithValue(
542     address target,
543     bytes memory data,
544     uint256 value,
545     string memory errorMessage
546 ) internal returns (bytes memory) {
547     require(address(this).balance >= value, "Address: insufficient balance for call");
548     require(isContract(target), "Address: call to non-contract");
549
550     (bool success, bytes memory returndata) = target.call(value: value)(data);
```

```
function transferOwnership(address newOwner) public onlyOwner {
function updateFees(uint256 _liqFee, uint256 _treasuryFee, uint256 _insuranceFee, uint256 _pitFee,
    uint256 _sellFee) external onlyOwner{
function updatetokenSwapAmountThreshold(uint256 _amount) external onlyOwner{
    function sendToTreasury() public payable onlyOwner {
    function withdrawAllToTreasury() external swapping onlyOwner {
function updatesFeeExempt(address _addr, bool _status) external onlyOwner {
    function setAutoRebase(bool _flag) external onlyOwner {
    function setAutoAddLiquidity(bool _flag) external onlyOwner {
function setFeeReceivers( address _autoLiquidityReceiver, address _treasuryReceiver, address _CATEPit)
    external onlyOwner {
    function setWhitelist(address _addr) external onlyOwner {
function setBotBlacklist(address _botAddress, bool _flag) external onlyOwner {
    (Owner can blacklist wallet)
function setPairAddress(address _pairAddress) public onlyOwner {
    function setLP(address _address) external onlyOwner {
```

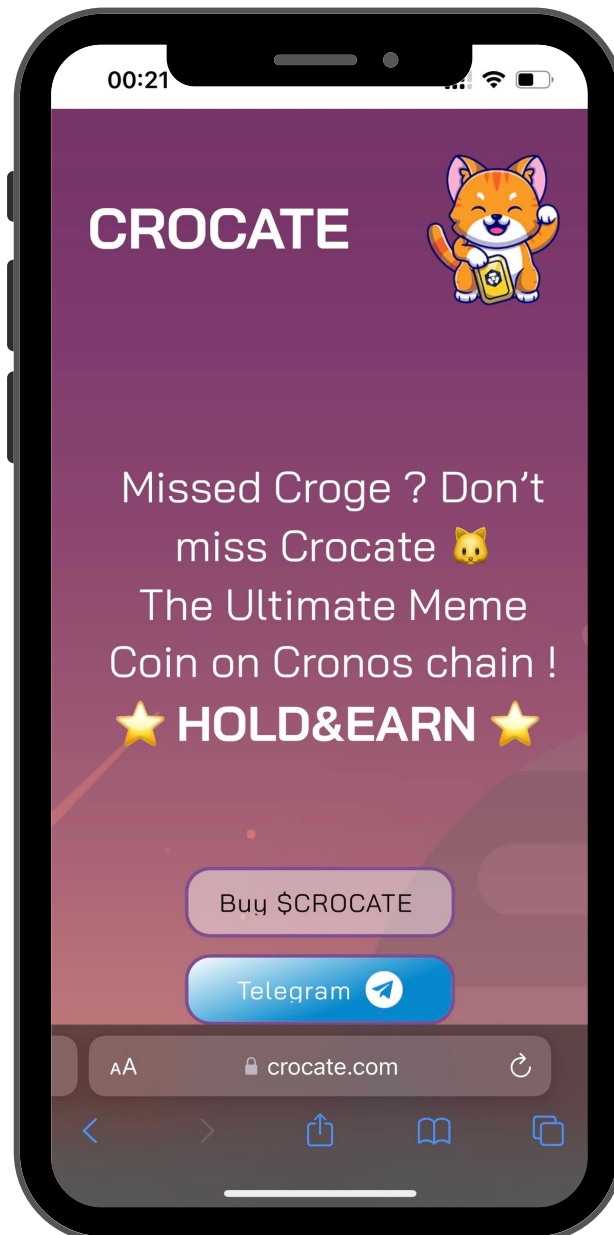
# Issue Checking:

| N° | Issue description.  | Checking Status. |
|----|---|------------------|
| 1  | Compiler Errors   | Passed           |
| 2  | Race conditions and Reentrancy. Cross-function race conditions. | Passed           |
| 3  | Possible delays in data delivery.                               | Passed           |
| 4  | Oracle calls.   | Passed           |
| 5  | Front running.  | Passed           |
| 6  | Timestamp dependence.   | Passed           |
| 7  | Integer Overflow and Underflow.                                 | Passed           |
| 8  | DoS with Revert.  | Passed           |
| 9  | DoS with block gas limit.                                       | Passed           |
| 10 | Methods execution permissions.                                  | Passed           |
| 11 | Economy model.  | Passed           |
| 12 | The impact of the exchange rate on the logic.                   | Passed           |
| 13 | Private user data leaks.  | Passed           |
| 14 | Malicious Event log   | Passed           |
| 15 | Scoping and Declarations.                                       | Passed           |
| 16 | Uninitialized storage pointers.                                 | Passed           |
| 17 | Arithmetic accuracy.  | Passed           |
| 18 | Design Logic.   | Passed           |
| 19 | Cross-function race conditions.                                 | Passed           |
| 20 | Safe Zeppelin module  | Passed           |
| 21 | Fallback function security.                                     | Passed           |

Audit Result:

**PASSED**

# WEBSITE - OVERVIEW



🕒 **Domain Registration: 2022-04-13**

1 day old

**SSL CERTIFICATE: A**

(<https://www.ssllabs.com>)

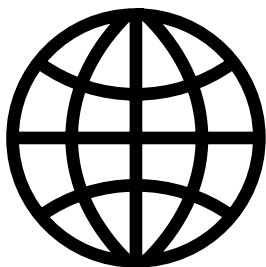
# **SOCIAL MEDIA - OVERVIEW**



**CRONOSCATÉ**



**CROCATÉ1**



**CROCATÉ.COM**